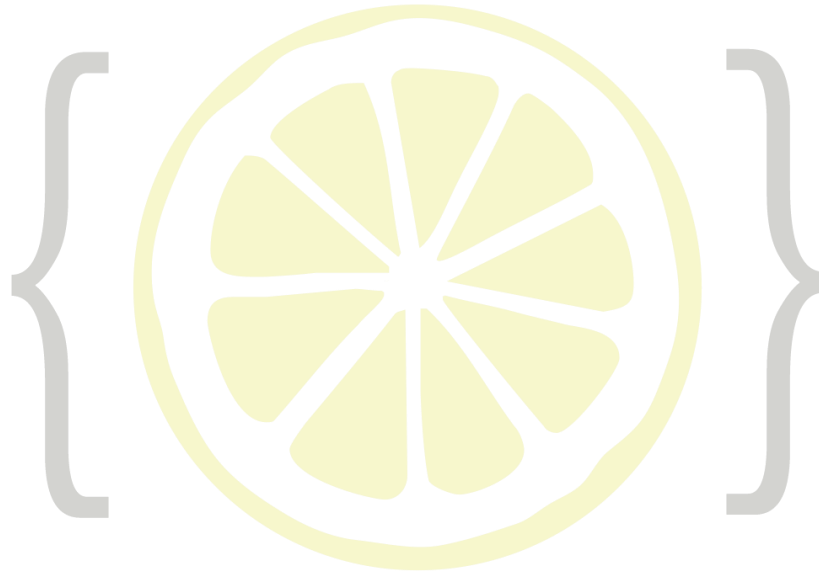


CSS

Especificidad, Cascada, Herencia

A decorative graphic in the bottom right corner of the slide, showing a portion of a lime slice with its characteristic green segments and white pith.

Agenda



Especificidad

"La especificidad es la manera mediante la cual los navegadores deciden qué valores de una propiedad CSS son más relevantes para un elemento y, por lo tanto, serán aplicados."

La especificidad se calcula en base a la jerarquía de los selectores y en función del orden y lugar donde estén ubicados los estilos

Aunque pensemos en términos de selectores y reglas, no es toda la regla la que se sobrescribe, sino solo las propiedades que entran en conflicto

Si dos reglas o propiedades en conflicto tienen la misma especificidad se aplicará siempre la de más abajo (la procesada más recientemente)

Jerarquía de mayor a menor

```
<a id="lemon" class="code" href="url">Lemoncode</a>
```

Elemento

Estilos inline. Se aplican en el propio elemento HTML mediante el atributo **style**. No recomendable.

```
<a style="width: 60px">Lemoncode</a>
```

Selectores ID. (identificadores) Son los selectores más específicos ya que hacen referencia a un único elemento dentro del documento.

```
#lemon {...styles}
```

Clases, atributos y pseudo-clases.

```
.code {...styles}  
a[href="url"] {...styles}  
a:hover {...styles}
```

Elementos y pseudo-elementos

```
a{...styles}  
a:first-letter {...styles}
```

La excepción **!important**

Cuando utilizamos **!important** en una propiedad esta sobrescribe a cualquier otra teniendo prioridad sobre todas las demás

```
p{ color: red !important; }
```



El uso de !important está considerado como una mala práctica



Su uso hace que el código sea más difícil de depurar



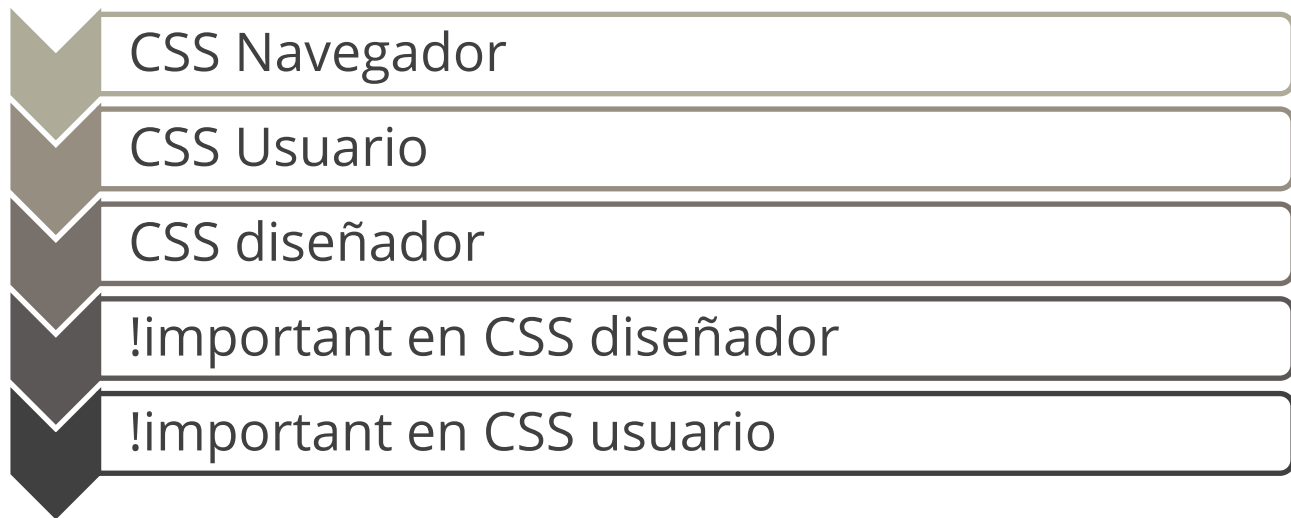
Utilízalo solo cuando realmente sea necesario



Cascada

CSS significa hojas de estilo en cascada (cascading style sheets). La cascada, y el concepto estrechamente relacionado de especificidad son mecanismos que controlan qué regla se aplica cuando aparecen conflictos. La cascada en las hojas de estilo significa que el orden de las reglas importa en CSS: cuando dos reglas tienen la misma especificidad, se aplica la que aparece en último lugar en el CSS

Prioridad de menos a más en las hojas de estilos





Herencia

El concepto de herencia en CSS significa que algunas propiedades heredan por defecto los valores establecidos en el elemento padre, pero otras no.

Las propiedades que se heredan por defecto y las que no, son cuestión en gran medida de sentido común.

CSS proporciona cuatro valores de propiedad universales especiales para el control de la herencia.

Control de herencia

Todas las propiedades CSS aceptan estos valores:

- **inherit** hace que la propiedad del elemento al cual se aplica tome el valor de la propiedad de su elemento padre.
- **initial** hace que la propiedad del elemento al cual se aplica tenga el mismo valor que esté establecido para esa propiedad en la hoja de estilo por defecto del navegador.
- **unset** restablece la propiedad a su valor natural, lo que significa que si la propiedad se hereda de forma natural, actúa como ``inherit``, y en caso contrario como ``initial``.
- **revert** restablece la propiedad a su valor heredado si hereda de su padre o al valor predeterminado establecido para esa propiedad en el elemento por el navegador o usuario. Se puede aplicar a cualquier propiedad CSS incluida la abreviatura ``all``.

Control de herencia

```
<style>
  div { font-weight: bold; }

  h3 { font-weight: lighter; }
</style>
```

<!-- Inherit en el h3 hace que su propiedad "font-weight" tome el valor del padre, es decir "bold" -->

```
<div>
  <h3 style="font-weight: inherit;"></h3>
</div>
```

<!-- Initial restablece el texto como "font-weight: normal" porque este es el valor por defecto de la propiedad "font-weight" -->

```
<h3 style="font-weight: initial;"></h3>
```

<!-- Unset restablece el texto como "font-weight: normal" porque este es el valor por defecto de la propiedad "font-weight" (unset en este caso actúa como initial) -->

```
<h3 style="font-weight: unset;"></h3>
```

<!-- Unset en el h3 hace que su propiedad "font-weight" tome el valor del padre, es decir "bold" (unset en este caso actúa como inherit) -->

```
<div>
  <h3 style="font-weight: unset;"></h3>
</div>
```

<!-- Revert restablece el texto como "font-weight: bold" porque este es el valor por defecto de la propiedad "font-weight" en el elemento "h3" -->

```
<h3 style="font-weight: revert;"></h3>
```

¡ Muchas gracias !



@lemoncoders



<https://github.com/lemoncode>



@basefactorteam